

REMARKS

By this Amendment, Applicants amend claims 1-3, 6-10, 13-17, 20, and 21; and cancel claims 4, 11, 18, and 22-26, without prejudice or disclaimer of the subject matter recited therein. Support for the claim amendments can be found, for example, in the specification, paras. 032, 044, 052-055 and in Figs. 1, 7-9. No prohibited new matter has been added. Claims 1-3, 5-10, 12-17, and 19-21 remain pending.

In the Office Action,¹ claims 22-26 are rejected under 35 U.S.C. § 101 as being directed to non-statutory subject matter; claims 1, 2, 6-9, 13-16, 20-22, and 26 are rejected under 35 U.S.C. § 102(b) as being anticipated by U.S. Patent No. 5,835,953 to Ohran ("Ohran"); and claims 3-5, 10-12, 17-19, and 23-25 are rejected under 35 U.S.C. § 103(a) as being unpatentable over *Ohran* in view of U.S. Patent No. 5,548,750 to Larsson et al. ("Larsson").

For the following reasons, Applicants traverse the rejections of the pending claims and submit that the pending claims are patentable over the prior art of record. The rejections of claims 4, 11, 18, and 22-26 are rendered moot by the cancellation of these claims.

I. Rejection of Claims 1, 2, 6-9, 13-16, 20, and 21 under 35 U.S.C. § 102(b)

Applicants respectfully traverse the rejection of claims 1, 2, 6-9, 13-16, 20, and 21 under 35 U.S.C. § 102(b) as being unpatentable over *Ohran*. In order to properly establish that *Ohran* anticipates Applicants' claims under 35 U.S.C. § 102, each and

¹ The Office Action contains a number of statements reflecting characterizations of the related art and the claims. Regardless of whether any such statement is identified herein, Applicants decline to automatically subscribe to any statement of characterization in the Office Action.

every element of each of the claims must be found, either expressly described or under principles of inherency, in that single reference. See M.P.E.P. § 2131 (8th ed., rev. 7, July 2008) (quoting *Verdegaal Bros. v. Union Oil Co. of California*, 814 F.2d 628 631, 2 U.S.P.Q.2d 1051, 1053 (Fed. Cir. 1987)). Furthermore, “[t]he identical invention must be shown in as complete detail as is contained in the . . . claim.” *Id.* (quoting *Richardson v. Suzuki Motor Co.*, 868 F.2d 1226, 1236, 9 U.S.P.Q.2d 1913, 1920 (Fed. Cir. 1989)). Applicants submit that *Ohran* does not teach or suggest each and every element of Applicants’ claims.

Independent claim 1 recites a method for controlling access to a data object comprising, for example, “determining, using a processor, whether another process is attempting to perform a transaction with the data object by determining whether the ID [assigned to the data object] is stored in the transactional lock object” and “upon determining that the ID is not stored in the transactional lock object and that another process is not attempting to perform a transaction with the data object, storing the ID in the transactional lock object.”

The Office Action alleges that column 5, lines 49-51 and column 11, lines 30-33 of *Ohran* disclose the above-quoted features of claim 1. Office Action at 3. This is incorrect.

Ohran relates to “keeping a map which identifies those storage locations that have new data written in them starting with time T_0 .” *Ohran*, col. 10, ll. 61-62. *Ohran* further discloses that “[t]he map . . . that was used to track which storage locations had data written therein between time T_0 and time T_1 is used to identify the data that should be transferred to the backup storage device.” *Ohran*, col. 11, ll. 31-34.

The map of *Ohran* does not teach or suggest the claimed “transactional lock object.” The map of *Ohran* tracks storage locations that have already been modified. *Ohran* fails to teach or suggest “determining . . . whether another process is attempting to perform a transaction with the data object by determining whether the ID is stored in a transaction lock object,” as recited in claim 1 (emphasis added).

Ohran discloses that “[b]ackup map 48 may comprise a boolean entry for each storage location on mass storage device 20. When a storage location has new data written in it, the entry for the storage location may then be set.” *Ohran*, col. 13, ll. 22-24. However, *Ohran* fails to teach or suggest setting the boolean entry for the storage location in the backup map “upon determining that the ID is not stored in the transactional lock object,” as recited in claim 1. In other words, even if the map could be considered to be some form of a “lock object,” *Ohran* fails to disclose making a determination regarding the state of the backup map before setting the boolean entry in the backup map.

Moreover, *Ohran* fails to teach or suggest setting the boolean entry for the storage location in the backup map “upon determining . . . that another process is not attempting to perform a transaction with the data object,” as recited in claim 1. In contrast, in *Ohran* the boolean entry for the storage location in the backup map is set if another process does modify the storage location.

For at least the above reasons, *Ohran* fails to teach or suggest the “transactional lock object” of claim 1.

The Office Action further alleges that item 22 in Fig. 7A of *Ohran* discloses the above-noted features of claim 1. Office Action at 3. This is incorrect.

Ohran discloses “copying a data block that is to be overwritten from mass storage device 20 into snapshot storage 22 and then indicating that the block has been preserved in snapshot map 52.” *Ohran*, col. 14, ll. 2-4. However, *Ohran* discloses that “it is preferred that the static snapshot be taken in such a way that user access to the primary mass storage device is not interrupted.” *Ohran*, col. 6, ll. 1-4 (emphasis added). Therefore, neither the snapshot storage 22 nor the snapshot map 52 of *Ohran* teach or suggest the “transactional lock object” of claim 1.

Furthermore, *Ohran* does not disclose storing a storage location in the snapshot storage 22 upon determining that the storage location is not stored in the snapshot storage 22. *Ohran* also does not disclose setting a boolean flag for a storage location in the snapshot map 52 upon determining that the boolean flag is not set. As a result, *Ohran* fails to teach or suggest “upon determining that the ID is not stored in the transactional lock object . . . , storing the ID in the transactional lock object,” as recited in claim 1.

For at least these additional reasons, *Ohran* fails to teach or suggest the “transactional lock object” of claim 1.

Accordingly, despite the teachings of a map, a snapshot storage, and a snapshot map, *Ohran* fails to teach or suggest “determining, using a processor, whether another process is attempting to perform a transaction with the data object by determining whether the ID [assigned to the data object] is stored in the transactional lock object” and “upon determining that the ID is not stored in the transactional lock object and that another process is not attempting to perform a transaction with the data object, storing the ID in the transactional lock object,” as recited in independent claim 1.

In addition, *Ohran* fails to teach or suggest other features of claim 1. For example, *Ohran* fails to teach or suggest “determining, using the processor, whether another process is archiving the data object to a second storage location by determining whether the ID is stored in a permanent lock object” and “upon determining that the ID is not stored in the permanent lock object and that another process is not archiving the data object to the second storage location, granting access to the data object.”

Ohran discloses several archiving techniques. However, *Ohran* is completely silent with respect to the claimed step of “determining . . . whether another process is archiving the data object,” as recited in claim 1. Therefore, *Ohran* fails to teach or suggest “determining . . . whether another process is archiving the data object to a second storage location by determining whether the ID is stored in a permanent lock object” and also fails to teach or suggest “granting access to the data object” “upon determining . . . that another process is not archiving the data object to the second storage location,” as recited in claim 1.

For at least the foregoing reasons, *Ohran* fails to anticipate claim 1. Furthermore, independent claims 8 and 15, although different in scope from independent claim 1, are allowable for at least reasons similar to those presented with respect to claim 1. Dependent claims 2, 6, 7, 9, 13, 14, 16, 20, and 21 are allowable at least due to their dependence from an allowable base claim.

Accordingly, Applicants respectfully request that the Examiner withdraw the rejection of claims 1, 2, 6-9, 13-16, 20, and 21 under 35 U.S.C. § 102(b), and allow these claims.

II. **Rejection of Claims 3, 5, 10, 12, 17, and 19 under 35 U.S.C. § 103(a)**

Applicants respectfully traverse the rejection of claims 3, 5, 10, 12, 17, and 19 under 35 U.S.C. § 103(a) as being unpatentable over *Ohran* in view of *Larsson*. A *prima facie* case of obviousness has not been established.

“The key to supporting any rejection under 35 U.S.C. 103 is the clear articulation of the reason(s) why the claimed invention would have been obvious.” M.P.E.P. § 2141(III). “[T]he framework for objective analysis for determining obviousness under 35 U.S.C. 103 is stated in *Graham v. John Deere Co.*, 383 U.S. 1, 148 USPQ 459 (1966). . . . The factual inquiries . . . are as follows:

(A) [Determining the scope and content of the prior art;]

(B) Ascertaining the differences between the claimed invention and the prior art;

and

(C) Resolving the level of ordinary skill in the pertinent art.”

M.P.E.P. § 2141(II). “Office personnel must explain why the difference(s) between the prior art and the claimed invention would have been obvious to one of ordinary skill in the art.” M.P.E.P. § 2141(III).

Claims 3 and 5 depend from and require all features of claim 1. As discussed above, *Ohran* fails to teach or suggest all the features of claim 1. Even if the Office Action’s characterizations of *Larsson* regarding features of claims 3 and 5 are correct, which they are not, *Larsson* also fails to teach or suggest all the above-noted features of claim 1, which are required by claims 3 and 5.

For example, claim 1 recites “storing the ID [assigned to the data object] in the transactional lock object.” *Larsson* discloses that the *BackupSynch* variable in the *LID*

table may be set to either “include” or “exclude.” *Larsson*, col. 8, ll. 7-10. *Larsson* further discloses that the LID table includes pointers to objects. *Larsson*, col. 4, lines 32-36. However, the LID table of *Larsson* does not store an ID of a data object. Furthermore, the BackupSynch variable in the LID table of *Larsson* also does not store an ID of a data object. Therefore, *Larsson* fails to teach or suggest the claimed “transactional lock object” and, more specifically, fails to teach or suggest “storing the ID [assigned to the data object] in the transactional lock object,” as recited in claim 1.

Furthermore, *Larsson* fails to teach or suggest “determining . . . whether another process is attempting to perform a transaction with the data object by determining whether the ID is stored in a transactional lock object,” as recited in claim 1.

Larsson discloses that “[c]opying objects to the backup area will . . . not start until the local data base handler has made all changes from transactions to be included in the backup.” *Larsson*, col. 6, ll. 15-18. However, the system of *Larsson* does not include the claimed “transactional lock object” and the “ID,” and, therefore, is incapable of “determining . . . whether another process is attempting to perform a transaction with the data object by determining whether the ID is stored in a transaction lock object,” as recited in claim 1.

Moreover, in *Larsson*, the only variable in the LID table is the BackupSynch variable, which can be set to “include” or “exclude.” This indicates whether a data object should be copied to the backup area, but *Larsson* does not “determin[e] whether the ID is stored in a transactional lock object,” as recited in claim 1. Accordingly, *Larsson* fails to teach or suggest “determining . . . whether another process is

attempting to perform a transaction with the data object by determining whether the ID is stored in a transactional lock object,” as recited in claim 1.

In addition, *Larsson* also fails to teach or suggest “upon determining that the ID is not stored in the transactional lock object and that another process is not attempting to perform a transaction with the data object, storing the ID in the transactional lock object,” as recited in claim 1. As discussed above, *Larsson* fails to teach or suggest “storing the ID [assigned to the data object] in the transactional lock object,” as recited in claim 1. As a result, *Larsson* also fails to teach or suggest “upon determining that the ID is not stored in the transactional lock object,” performing the step of “storing the ID in the transactional lock object,” as recited in claim 1. The ability to detect the storage of an ID in a lock object or to store such an ID in a lock object is completely absent in *Larsson*.

Furthermore, *Larsson* fails to teach or suggest “determining whether another process is archiving the data object to a second storage location by determining whether the ID is stored in a permanent lock object” and “upon determining that the ID is not stored in the permanent lock object and that another process is not archiving the data object to the second storage location, granting access to the data object,” as recited in claim 1. In *Larsson*, granting access to the data object to be copied into the backup area is dependent on the BackupSynch variable in the LID table, but *Larsson* does not “determin[e] . . . whether another process is archiving the data object to a second storage location . . . [before] storing the ID in the permanent lock object,” as recited in claim 1.

Moreover, *Larsson* teaches away from the method of claim 1. *Larsson* discloses that “transactions are allowed to perform operations to the data base at the same time as backing up is performed” and “transactions shall be allowed to perform operations towards the data base simultaneously with backing up.” *Larsson*, col. 2, ll. 12-13; col. 4, ll. 56-59. In the “Description of the Related Art” section of Applicants’ specification, Applicants address several problems that can arise if data is modified during a backup. See, e.g., Specification, paras. 006-008. These problems can occur in *Larsson*, because transactions are allowed to perform operations towards the data base simultaneously with backing up, but *Larsson* does not solve or even address these problems. In contrast, claim 1 recites “determining . . . whether another process is attempting to perform a transaction with the data object” and “determining . . . whether another process is archiving the data object to a second storage location.” *Larsson*’s use of a central backup handler negates the need for such determining steps, and *Larsson* explicitly teaches away from the claimed determining steps by stating that “transactions [are] allowed to perform operations towards the data base simultaneously with backing up.” *Larsson*, col. 4, ll. 56-58.

For the reasons discussed above, *Larsson* fails to cure all the deficiencies of *Ohran* and, therefore, a *prima facie* case of obviousness has not been established with respect to claims 3 and 5. Independent claims 10, 12, 17, and 19, although different in scope from claims 3 and 5, are allowable for at least reasons similar to those presented above with respect to claims 3 and 5. Accordingly, Applicants respectfully request that the Examiner withdraw the rejection of claims 3, 5, 10, 12, 17, and 19 under 35 U.S.C. § 103(a), and allow these claims.

CONCLUSION

In view of the foregoing, Applicants respectfully request reconsideration of this application and timely allowance of the pending claims.

Please grant any extensions of time required to enter this response and charge any additional required fees to our deposit account 06-0916.

Respectfully submitted,

FINNEGAN, HENDERSON, FARABOW,
GARRETT & DUNNER, L.L.P.

Dated: April 21, 2009

By: 

Peter C. Yi
Reg. No. 61,790
202.408.4485